Network Working Group
Request for Comments: 3794
Category: Informational

P. Nesser, II
Nesser & Nesser Consulting
A. Bergstrom, Ed.
Ostfold University College

June 2004

Survey of IPv4 Addresses in Currently Deployed IETF Transport Area Standards Track and Experimental Documents

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This document seeks to document all usage of IPv4 addresses in currently deployed IETF Transport Area documented standards. In order to successfully transition from an all IPv4 Internet to an all IPv6 Internet, many interim steps will be taken. One of these steps is the evolution of current protocols that have IPv4 dependencies. It is hoped that these protocols (and their implementations) will be redesigned to be network address independent, but failing that will at least dually support IPv4 and IPv6. To this end, all Standards (Full, Draft, and Proposed) as well as Experimental RFCs will be surveyed and any dependencies will be documented.

Table of Contents

1.0. Introduction 2
2.0. Document Organization. 2
3.0. Full Standards 2
4.0. Draft Standards. 10
5.0. Proposed Standards 11
6.0. Experimental RFCs. 22
7.0. Summary of Results 27
    7.1. Standards. 27
    7.2. Draft Standards. 27
    7.3. Proposed Standards 27
    7.4. Experimental RFCs. 29
8.0. Security Considerations. 30
9.0. Acknowledgements 30

10.0. Normative Reference. 30
11.0. Authors' Addresses 30
12.0. Full Copyright Statement 31

1.0. Introduction

This document is part of a document set aiming to document all usage of IPv4 addresses in IETF standards. In an effort to have the information in a manageable form, it has been broken into 7 documents conforming to the current IETF areas (Application, Internet, Operations & Management, Routing, Security, Sub-IP and Transport).

For a full introduction, please see the introduction [1].

2.0. Document Organization

The rest of the document sections are described below.

Sections 3, 4, 5, and 6 each describe the raw analysis of Full, Draft, and Proposed Standards, and Experimental RFCs. Each RFC is discussed in its turn starting with RFC 1 and ending with (around) RFC 3100. The comments for each RFC are "raw" in nature. That is, each RFC is discussed in a vacuum and problems or issues discussed do not "look ahead" to see if the problems have already been fixed.

Section 7 is an analysis of the data presented in Sections 3, 4, 5, and 6. It is here that all of the results are considered as a whole and the problems that have been resolved in later RFCs are correlated.

3.0. Full Standards

Full Internet Standards (most commonly simply referred to as "Standards") are fully mature protocol specification that are widely implemented and used throughout the Internet.

3.1. RFC 768 User Datagram Protocol

Although UDP is a transport protocol there is one reference to the UDP/IP interface that states; "The UDP module must be able to determine the source and destination internet addresses and the protocol field from the internet header." This does not force a rewrite of the protocol but will clearly cause changes in implementations.

3.2. RFC 793 Transmission Control Protocol

Section 3.1 which specifies the header format for TCP. The TCP header is free from IPv4 references but there is an inconsistency in the computation of checksums. The text says: "The checksum also covers a 96 bit pseudo header conceptually prefixed to the TCP header. This pseudo header contains the Source Address, the Destination Address, the Protocol, and TCP length." The first and second 32-bit words are clearly meant to specify 32-bit IPv4 addresses. While no modification of the TCP protocol is necessitated by this problem, an alternate needs to be specified as an update document, or as part of another IPv6 document.

3.3. RFC 907 Host Access Protocol specification

This is a layer 3 protocol, and has as such no IPv4 dependencies.

3.4. NetBIOS Service Protocols. RFC1001, RFC1002

3.4.1. RFC 1001 PROTOCOL STANDARD FOR A NetBIOS SERVICE ON A TCP/UDP TRANSPORT: CONCEPTS AND METHODS

Section 15.4.1. RELEASE BY B NODES defines:

A NAME RELEASE DEMAND contains the following information:

- NetBIOS name
- The scope of the NetBIOS name
- Name type: unique or group
- IP address of the releasing node
- Transaction ID

Section 15.4.2. RELEASE BY P NODES defines:

A NAME RELEASE REQUEST contains the following information:

- NetBIOS name
- The scope of the NetBIOS name
- Name type: unique or group
- IP address of the releasing node
- Transaction ID

A NAME RELEASE RESPONSE contains the following information:

- NetBIOS name
- The scope of the NetBIOS name
- Name type: unique or group
- IP address of the releasing node
- Transaction ID
- Result:
  - Yes: name was released
  - No: name was not released, a reason code is provided

Section 16. NetBIOS SESSION SERVICE states:

The NetBIOS session service begins after one or more IP addresses have been found for the target name. These addresses may have been acquired using the NetBIOS name query transactions or by other means, such as a local name table or cache.

Section 16.1. OVERVIEW OF NetBIOS SESSION SERVICE

Session service has three phases:

Session establishment - it is during this phase that the IP address and TCP port of the called name is determined, and a TCP connection is established with the remote party.

6.1.1. SESSION ESTABLISHMENT PHASE OVERVIEW

An end-node begins establishment of a session to another node by somehow acquiring (perhaps using the name query transactions or a local cache) the IP address of the node or nodes purported to own the destination name.

Once the TCP connection is open, the calling node sends session service request packet. This packet contains the following information:

- Calling IP address (see note)
- Calling NetBIOS name
- Called IP address (see note)
- Called NetBIOS name

NOTE: The IP addresses are obtained from the TCP service interface.

If a compatible LISTEN exists, and there are adequate resources, then the session server may transform the existing TCP connection into the NetBIOS data session. Alternatively, the session server may redirect, or "retarget" the caller to another TCP port (and IP address).

If the caller is redirected, the caller begins the session establishment anew, but using the new IP address and TCP port given in the retarget response. Again a TCP connection is created, and again the calling and called node exchange credentials. The called party may accept the call, reject the call, or make a further redirection.

17.1. OVERVIEW OF NetBIOS DATAGRAM SERVICE

Every NetBIOS datagram has a named destination and source. To transmit a NetBIOS datagram, the datagram service must perform a name query operation to learn the IP address and the attributes of the destination NetBIOS name. (This information may be cached to avoid the overhead of name query on subsequent NetBIOS datagrams.)

17.1.1. UNICAST, MULTICAST, AND BROADCAST

NetBIOS datagrams may be unicast, multicast, or broadcast. A NetBIOS datagram addressed to a unique NetBIOS name is unicast. A NetBIOS datagram addressed to a group NetBIOS name, whether there are zero, one, or more actual members, is multicast. A NetBIOS datagram sent using the NetBIOS "Send Broadcast Datagram" primitive is broadcast.

17.1.2. FRAGMENTATION OF NetBIOS DATAGRAMS

When the header and data of a NetBIOS datagram exceeds the maximum amount of data allowed in a UDP packet, the NetBIOS datagram must be fragmented before transmission and reassembled upon receipt.

A NetBIOS Datagram is composed of the following protocol elements:

- IP header of 20 bytes (minimum)
- UDP header of 8 bytes
- NetBIOS Datagram Header of 14 bytes
- The NetBIOS Datagram data.

18. NODE CONFIGURATION PARAMETERS

- B NODES:
  - Node's permanent unique name
  - Whether IGMP is in use
  - Broadcast IP address to use
  - Whether NetBIOS session keep-alives are needed
  - Usable UDP data field length (to control fragmentation)
- P NODES:
  - Node's permanent unique name
  - IP address of NBNS
  - IP address of NBDD
  - Whether NetBIOS session keep-alives are needed
  - Usable UDP data field length (to control fragmentation)
- M NODES:
  - Node's permanent unique name
  - Whether IGMP is in use
  - Broadcast IP address to use
  - IP address of NBNS
  - IP address of NBDD
  - Whether NetBIOS session keep-alives are needed
  - Usable UDP data field length (to control fragmentation)

All of the proceeding sections make implicit use of IPv4 addresses and a new specification should be defined for use of IPv6 underlying addresses.

3.4.2. RFC 1002 PROTOCOL STANDARD FOR A NetBIOS SERVICE ON A TCP/UDP TRANSPORT: DETAILED SPECIFICATIONS

Section 4.2.1.3. RESOURCE RECORD defines

RESOURCE RECORD RR_TYPE field definitions:

Symbol	Value	Description:
A	0x0001	IP address Resource Record (See REDIRECT NAME QUERY RESPONSE)

Sections 4.2.2. NAME REGISTRATION REQUEST, 4.2.3. NAME OVERWRITE REQUEST & DEMAND, 4.2.4. NAME REFRESH REQUEST, 4.2.5. POSITIVE NAME REGISTRATION RESPONSE, 4.2.6. NEGATIVE NAME REGISTRATION RESPONSE, 4.2.7. END-NODE CHALLENGE REGISTRATION RESPONSE, 4.2.9. NAME RELEASE REQUEST & DEMAND, 4.2.10. POSITIVE NAME RELEASE RESPONSE, 4.2.11. NEGATIVE NAME RELEASE RESPONSE and Sections 4.2.13. POSITIVE NAME QUERY

RESPONSE all contain 32 bit fields labeled "NB_ADDRESS" clearly defined for IPv4 addresses Sections 4.2.15. REDIRECT NAME QUERY RESPONSE contains a field "NSD_IP_ADDR" which also is designed for a IPv4 address.

Section 4.3.5. SESSION RETARGET RESPONSE PACKET

```
                     1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      TYPE     |     FLAGS     |            LENGTH             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      RETARGET_IP_ADDRESS                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           PORT                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Section 4.4.1. NetBIOS DATAGRAM HEADER

```
                     1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   MSG_TYPE    |     FLAGS     |           DGM_ID              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           SOURCE_IP                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          SOURCE_PORT          |          DGM_LENGTH           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         PACKET_OFFSET         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

      Section 4.4.2. DIRECT_UNIQUE, DIRECT_GROUP, & BROADCAST
                      DATAGRAM

```
                     1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   MSG_TYPE    |     FLAGS     |           DGM_ID              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           SOURCE_IP                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          SOURCE_PORT          |          DGM_LENGTH           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         PACKET_OFFSET         |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               |
|                                                               |
/                          SOURCE_NAME                          /
/                                                               /
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
/                       DESTINATION_NAME                        /
/                                                               /
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
/                           USER_DATA                           /
/                                                               /
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

      Section 4.4.3. DATAGRAM ERROR PACKET

```
                     1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   MSG_TYPE    |     FLAGS     |           DGM_ID              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           SOURCE_IP                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          SOURCE_PORT          |  ERROR_CODE   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Section 4.4.4. DATAGRAM QUERY REQUEST

```
                     1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   MSG_TYPE    |     FLAGS     |           DGM_ID              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           SOURCE_IP                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          SOURCE_PORT          |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                                                               |
/                       DESTINATION_NAME                        /
/                                                               /
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.4.5. DATAGRAM POSITIVE AND NEGATIVE QUERY RESPONSE

```
                     1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   MSG_TYPE    |     FLAGS     |           DGM_ID              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           SOURCE_IP                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          SOURCE_PORT          |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                                                               |
/                       DESTINATION_NAME                        /
/                                                               /
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

5.3. NetBIOS DATAGRAM SERVICE PROTOCOLS

The following are GLOBAL variables and should be NetBIOS user configurable:

- BROADCAST_ADDRESS: the IP address B-nodes use to send datagrams with group name destinations and broadcast datagrams. The default is the IP broadcast address for a single IP network.

There is also a large amount of pseudo code for most of the protocols functionality that make no specific reference to IPv4 addresses. However they assume the use of the above defined packets. The pseudo code may be valid for IPv6 as long as the packet formats are updated.

3.5. RFC 1006 ISO Transport Service on top of the TCP (Version: 3)

Section 5. The Protocol defines a mapping specification

Mapping parameters is also straight-forward:

```
    network service             TCP
            -------             ---
                CONNECTION RELEASE

      Called address          server’s IP address
                              (4 octets)

      Calling address         client’s IP address
                              (4 octets)
```

4.0. Draft Standards

Draft Standards represent the penultimate standard level in the IETF. A protocol can only achieve draft standard when there are multiple, independent, interoperable implementations. Draft Standards are usually quite mature and widely used.

4.1. RFC 3530 Network File System (NFS) version 4 Protocol

There are no IPv4 dependencies in this specification.

4.2. RFC 3550 RTP: A Transport Protocol for Real-Time Applications

There are no IPv4 dependencies in this specification.

4.3. RFC 3551 RTP Profile for Audio and Video Conferences with Minimal Control.

There are no IPv4 dependencies in this specification.

5.0. Proposed Standards

Proposed Standards are introductory level documents. There are no requirements for even a single implementation. In many cases Proposed are never implemented or advanced in the IETF standards process. They therefore are often just proposed ideas that are presented to the Internet community. Sometimes flaws are exposed or they are one of many competing solutions to problems. In these later cases, no discussion is presented as it would not serve the purpose of this discussion.

5.01. RFC 1144 Compressing TCP/IP headers for low-speed serial links

This RFC is specifically oriented towards TCP/IPv4 packet headers and will not work in it’s current form. Significant work has already been done on similar algorithms for TCP/IPv6 headers.

5.02. RFC 1323 TCP Extensions for High Performance

There are no IPv4 dependencies in this specification.

5.03. RFC 1553 Compressing IPX Headers Over WAN Media (CIPX)

There are no IPv4 dependencies in this specification.

5.04. RFC 1692 Transport Multiplexing Protocol (TMux)

Section 6. Implementation Notes is states:

> Because the TMux mini-header does not contain a TOS field, only segments with the same IP TOS field should be contained in a single TMux message. As most systems do not use the TOS feature, this is not a major restriction. Where the TOS field is used, it may be desirable to hold several messages under construction for a host, one for each TOS value.
>
> Segments containing IP options should not be multiplexed.

This is clearly IPv4 specific, but a simple restatement in IPv6 terms will allow complete functionality.

5.05. RFC 1831 RPC: Remote Procedure Call Protocol Specification Version 2 RPC

There are no IPv4 dependencies in this specification.

5.06. RFC 1833 Binding Protocols for ONC RPC Version 2

In Section 2.1 RPCBIND Protocol Specification (in RPC Language) there is the following code fragment:

```
 * Protocol family (r_nc_protofmly):
 *   This identifies the family to which the protocol belongs.
 *   The following values are defined:
 *     NC_NOPROTOFMLY   "-"
 *     NC_LOOPBACK      "loopback"
 *     NC_INET          "inet"
 *     NC_IMPLINK       "implink"
 *     NC_PUP           "pup"
 *     NC_CHAOS         "chaos"
 *     NC_NS            "ns"
 *     NC_NBS           "nbs"
 *     NC_ECMA          "ecma"
 *     NC_DATAKIT       "datakit"
 *     NC_CCITT         "ccitt"
 *     NC_SNA           "sna"
 *     NC_DECNET        "decnet"
 *     NC_DLI           "dli"
 *     NC_LAT           "lat"
 *     NC_HYLINK        "hylink"
 *     NC_APPLETALK     "appletalk"
 *     NC_NIT           "nit"
 *     NC_IEEE802       "ieee802"
 *     NC_OSI           "osi"
 *     NC_X25           "x25"
 *     NC_OSINET        "osinet"
 *     NC_GOSIP         "gosip"
```

It is clear that the value for NC_INET is intended for the IP protocol and is seems clear that it is IPv4 dependent.

5.07. RFC 1962 The PPP Compression Control Protocol (CCP)

There are no IPv4 dependencies in this specification.

5.08. RFC 2018 TCP Selective Acknowledgement Options

There are no IPv4 dependencies in this specification.

5.09. RFC 2029 RTP Payload Format of Sun’s CellB Video Encoding

There are no IPv4 dependencies in this specification.

5.10. RFC 2032 RTP Payload Format for H.261 Video Streams

There are no IPv4 dependencies in this specification.

5.11. RFC 2126 ISO Transport Service on top of TCP (ITOT)

This specification is IPv6 aware and has no issues.

5.12. RFC 2190 RTP Payload Format for H.263 Video Streams

There are no IPv4 dependencies in this specification.

5.13. RFC 2198 RTP Payload for Redundant Audio Data

There are no IPv4 dependencies in this specification.

5.14. RFC 2205 Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification

In Section 1. Introduction the statement is made:

> RSVP operates on top of IPv4 or IPv6, occupying the place of a transport protocol in the protocol stack.

Appendix A defines all of the header formats for RSVP and there are multiple formats for both IPv4 and IPv6.

There are no IPv4 dependencies in this specification.

5.15. RFC 2207 RSVP Extensions for IPSEC Data Flows

The defined IPsec extensions are valid for both IPv4 & IPv6. There are no IPv4 dependencies in this specification.

5.16. RFC 2210 The Use of RSVP with IETF Integrated Services

There are no IPv4 dependencies in this specification.

5.17. RFC 2211 Specification of the Controlled-Load Network Element Service

There are no IPv4 dependencies in this specification.

5.18. RFC 2212 Specification of Guaranteed Quality of Service

There are no IPv4 dependencies in this specification.

5.19. RFC 2215 General Characterization Parameters for Integrated Service Network Elements

There are no IPv4 dependencies in this specification.

5.20. RFC 2250 RTP Payload Format for MPEG1/MPEG2 Video

There are no IPv4 dependencies in this specification.

5.21. RFC 2326 Real Time Streaming Protocol (RTSP)

Section 3.2 RTSP URL defines:

The "rtsp" and "rtspu" schemes are used to refer to network resources via the RTSP protocol. This section defines the scheme-specific syntax and semantics for RTSP URLs.

```
rtsp_URL  =   ( "rtsp:" | "rtspu:" )
              "//" host [ ":" port ] [ abs_path ]
host      =   <A legal Internet host domain name of IP
              address (in dotted decimal form), as defined
              by Section 2.1 of RFC 1123 \cite{rfc1123}>
port      =   *DIGIT
```

Although later in that section the following text is added:

The use of IP addresses in URLs SHOULD be avoided whenever possible (see RFC 1924 [19]).

Some later examples show:

Example:

```
C->S: DESCRIBE rtsp://server.example.com/fizzle/foo RTSP/1.0
      CSeq: 312
      Accept: application/sdp, application/rtsl,
              application/mheg

S->C: RTSP/1.0 200 OK
      CSeq: 312
      Date: 23 Jan 1997 15:35:06 GMT
      Content-Type: application/sdp
      Content-Length: 376

      v=0
      o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
      s=SDP Seminar
      i=A Seminar on the session description protocol
```

```
u=http://www.cs.ucl.ac.uk/staff/M.Handley/sdp.03.ps
e=mjh@isi.edu (Mark Handley)
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
a=recvonly
m=audio 3456 RTP/AVP 0
m=video 2232 RTP/AVP 31
m=whiteboard 32416 UDP WB
a=orient:portrait
```

which implies the use of the "IP4" tag and it should be possible to use an "IP6" tag. There are also numerous other similar examples using the "IP4" tag.

RTSP is also dependent on IPv6 support in a protocol capable of describing media configurations, for example SDP RFC 2327.

RTSP can be used over IPv6 as long as the media description protocol supports IPv6, but only for certain restricted use cases. For full functionality there is need for IPv6 support. The amount of updates needed are small.

5.22. RFC 2327 SDP: Session Description Protocol (SDP)

This specification is under revision, and IPv6 support was added in RFC 3266 which updates this specification.

5.23. RFC 2380 RSVP over ATM Implementation Requirements

This specification is both IPv4 and IPv6 aware.

5.24. RFC 2381 Interoperation of Controlled-Load Service and Guaranteed Service with ATM

There does not seem any inherent IPv4 limitations in this specification, but it assumes work of other standards that have IPv4 limitations.

5.25. RFC 2429 RTP Payload Format for the 1998 Version of ITU-T Rec. H.263 Video (H.263+)

There are no IPv4 dependencies in this specification.

5.26. RFC 2431 RTP Payload Format for BT.656 Video Encoding

There are no IPv4 dependencies in this specification.

5.27. RFC 2435 RTP Payload Format for JPEG-compressed Video

There are no IPv4 dependencies in this specification.

5.28. RFC 2474 Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers

This specification is both IPv4 and IPv6 aware.

5.29. RFC 2508 Compressing IP/UDP/RTP Headers for Low-Speed Serial Links

This specification is both IPv4 and IPv6 aware.

5.30. RFC 2581 TCP Congestion Control

There are no IPv4 dependencies in this specification.

5.31. RFC 2597 Assured Forwarding PHB Group

This specification is both IPv4 and IPv6 aware.

5.32. RFC 2658 RTP Payload Format for PureVoice(tm) Audio

There are no IPv4 dependencies in this specification.

5.33. RFC 2678 IPPM Metrics for Measuring Connectivity

This specification only supports IPv4.

5.34. RFC 2679 A One-way Delay Metric for IPPM

This specification only supports IPv4.

5.35. RFC 2680 A One-way Packet Loss Metric for IPPM

This specification only supports IPv4.

5.36. RFC 2681 A Round-trip Delay Metric for IPPM

This specification only supports IPv4.

5.37. RFC 2730 Multicast Address Dynamic Client Allocation Protocol (MADCAP)

This specification is both IPv4 and IPv6 aware and needs no changes.

5.38. RFC 2733 An RTP Payload Format for Generic Forward Error Correction

This specification is dependent on SDP which has IPv4 dependencies. Once that limitation is fixed, then this specification should support IPv6.

5.39. RFC 2745 RSVP Diagnostic Messages

This specification is both IPv4 and IPv6 aware and needs no changes.

5.40. RFC 2746 RSVP Operation Over IP Tunnels

This specification is both IPv4 and IPv6 aware and needs no changes.

5.41. RFC 2750 RSVP Extensions for Policy Control

There are no IPv4 dependencies in this specification.

5.42. RFC 2793 RTP Payload for Text Conversation

There are no IPv4 dependencies in this specification.

5.43. RFC 2814 SBM (Subnet Bandwidth Manager): A Protocol for RSVP-based Admission Control over IEEE 802-style networks

This specification claims to be both IPv4 and IPv6 aware, but all of the examples are given with IPv4 addresses. That, by itself is not a telling point but the following statement is made:

> a) LocalDSBMAddrInfo -- current DSBM’s IP address (initially, 0.0.0.0) and priority. All IP addresses are assumed to be in network byte order. In addition, current DSBM’s L2 address is also stored as part of this state information.

which could just be sloppy wording. Perhaps a short document clarifying the text is appropriate.

5.44. RFC 2815 Integrated Service Mappings on IEEE 802 Networks

There are no IPv4 dependencies in this specification.

5.45. RFC 2833 RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals

There are no IPv4 dependencies in this specification.

5.46. RFC 2848 The PINT Service Protocol: Extensions to SIP and SDP for IP Access to Telephone Call Services

This specification is dependent on SDP which has IPv4 dependencies. Once these limitations are fixed, then this specification should support IPv6.

5.47. RFC 2862 RTP Payload Format for Real-Time Pointers

There are no IPv4 dependencies in this specification.

5.48. RFC 2872 Application and Sub Application Identity Policy Element for Use with RSVP

There are no IPv4 dependencies in this specification.

5.49. RFC 2873 TCP Processing of the IPv4 Precedence Field

This specification documents a technique using IPv4 headers. A similar technique, if needed, will need to be defined for IPv6.

5.50. RFC 2883 An Extension to the Selective Acknowledgement (SACK) Option for TCP

There are no IPv4 dependencies in this specification.

5.51. RFC 2907 MADCAP Multicast Scope Nesting State Option

This specification is both IPv4 and IPv6 aware and needs no changes.

5.52. RFC 2960 Stream Control Transmission Protocol

This specification is both IPv4 and IPv6 aware and needs no changes.

5.53. RFC 2961 RSVP Refresh Overhead Reduction Extensions

This specification is both IPv4 and IPv6 aware and needs no changes.

5.54. RFC 2976 The SIP INFO Method

There are no IPv4 dependencies in this specification.

5.55. RFC 2988 Computing TCP’s Retransmission Timer

There are no IPv4 dependencies in this specification.

5.56. RFC 2996 Format of the RSVP DCLASS Object

There are no IPv4 dependencies in this specification.

5.57. RFC 2997 Specification of the Null Service Type

There are no IPv4 dependencies in this specification.

5.58. RFC 3003 The audio/mpeg Media Type

There are no IPv4 dependencies in this specification.

5.59. RFC 3006 Integrated Services in the Presence of Compressible Flows

This document defines a protocol that discusses compressible flows, but only in an IPv4 context. When IPv6 compressible flows are defined, a similar technique should also be defined.

5.60. RFC 3016 RTP Payload Format for MPEG-4 Audio/Visual Streams

There are no IPv4 dependencies in this specification.

5.61. RFC 3033 The Assignment of the Information Field and Protocol Identifier in the Q.2941 Generic Identifier and Q.2957 User-to-user Signaling for the Internet Protocol

This specification is both IPv4 and IPv6 aware and needs no changes.

5.62. RFC 3042 Enhancing TCP's Loss Recovery Using Limited Transmit

There are no IPv4 dependencies in this specification.

5.63. RFC 3047 RTP Payload Format for ITU-T Recommendation G.722.1

There are no IPv4 dependencies in this specification.

5.64. RFC 3057 ISDN Q.921-User Adaptation Layer

There are no IPv4 dependencies in this specification.

5.65. RFC 3095 Robust Header Compression (ROHC): Framework and four profiles

This specification is both IPv4 and IPv6 aware and needs no changes.

5.66. RFC 3108 Conventions for the use of the Session Description Protocol (SDP) for ATM Bearer Connections

This specification is currently limited to IPv4 as amplified below:

> The range and format of the <rtcpPortNum> and <rtcpIPaddr> subparameters is per [1]. The <rtcpPortNum> is a decimal number between 1024 and 65535. It is an odd number. If an even number in this range is specified, the next odd number is used. The <rtcpIPaddr> is expressed in the usual dotted decimal IP address representation, from 0.0.0.0 to 255.255.255.255.

and

```
      <rtcpIPaddr>      IP address for  receipt  Dotted decimal,
                        7-15 chars of RTCP packets
```

5.67. RFC 3119 A More Loss-Tolerant RTP Payload Format for MP3 Audio

There are no IPv4 dependencies in this specification.

5.68. RFC 3124 The Congestion Manager

This document is IPv4 limited since it uses the IPv4 TOS header field.

5.69. RFC 3140 Per Hop Behavior Identification Codes

There are no IPv4 dependencies in this specification.

5.70. RFC 3173 IP Payload Compression Protocol (IPComp)

There are no IPv4 dependencies in this specification.

5.71. RFC 3181 Signaled Preemption Priority Policy Element

There are no IPv4 dependencies in this specification.

5.72. RFC 3182 Identity Representation for RSVP

There are no IPv4 dependencies in this specification.

5.73. RFC 3246 An Expedited Forwarding PHB (Per-Hop Behavior)

There are no IPv4 dependencies in this specification.

5.74. RFC 3261 SIP: Session Initiation Protocol

There are no IPv4 dependencies in this specification.

5.75. RFC 3262 Reliability of Provisional Responses in Session Initiation Protocol (SIP)

There are no IPv4 dependencies in this specification.

5.76. RFC 3263 Session Initiation Protocol (SIP): Locating SIP Servers

There are no IPv4 dependencies in this specification.

5.77. RFC 3264 An Offer/Answer Model with Session Description Protocol (SDP)

There are no IPv4 dependencies in this specification.

5.78. RFC 3265 Session Initiation Protocol (SIP)-Specific Event Notification

There are no IPv4 dependencies in this specification.

5.79. RFC 3390 Increasing TCP’s Initial Window

There are no IPv4 dependencies in this specification.

5.80. RFC 3525 Gateway Control Protocol Version 1

There are no IPv4 dependencies in this specification.

5.81. RFC 3544 IP Header Compression over PPP

There are no IPv4 dependencies in this specification.

6.0. Experimental RFCs

Experimental RFCs typically define protocols that do not have widescale implementation or usage on the Internet. They are often propriety in nature or used in limited arenas. They are documented to the Internet community in order to allow potential interoperability or some other potential useful scenario. In a few cases they are presented as alternatives to the mainstream solution to an acknowledged problem.

6.1. RFC 908 Reliable Data Protocol (RDP)

This document is IPv4 limited as stated in the following section:

4.1. IP Header Format

When used in the internet environment, RDP segments are sent using the version 4 IP header as described in RFC791, "Internet Protocol." The RDP protocol number is ??? (decimal). The time-to-live field should be set to a reasonable value for the network.

All other fields should be set as specified in RFC-791.

A new protocol specification would be needed to support IPv6.

6.02. RFC 938 Internet Reliable Transaction Protocol functional and interface specification (IRTP)

This specification states:

4.1. State Variables

Each IRTP is associated with a single internet address. The synchronization mechanism of the IRTP depends on the requirement that each IRTP module knows the internet addresses of all modules with which it will communicate. For each remote internet address, an IRTP module must maintain the following information (called the connection table):

rem_addr (32 bit remote internet address)

A new specification that is IPv6 aware would need to be created.

6.03. RFC 998 NETBLT: A bulk data transfer protocol

This RFC states:

> The active end specifies a passive client through a client-specific "well-known" 16 bit port number on which the passive end listens. The active end identifies itself through a 32 bit Internet address and a unique 16 bit port number.

Clearly, this is IPv4 dependent, but could easily be modified to support IPv6 addressing.

6.04. RFC 1045 VMTP: Versatile Message Transaction Protocol

This specification has many IPv4 dependencies in its implementation appendices. For operations over IPv6 a similar implementation procedure must be defined. The IPv4 specific information is show below.

IV.1. Domain 1

For initial use of VMTP, we define the domain with Domain identifier 1 as follows:

```
+-----------+---------------+-----------------------+
| TypeFlags | Discriminator |    Internet Address   |
+-----------+---------------+-----------------------+
   4 bits         28 bits                32 bits
```

The Internet address is the Internet address of the host on which this entity-id is originally allocated. The Discriminator is an arbitrary value that is unique relative to this Internet host address. In addition, the host must guarantee that this identifier does not get reused for a long period of time after it becomes invalid. ("Invalid" means that no VMTP module considers in bound to an entity.) One technique is to use the lower order bits of a 1 second clock. The clock need not represent real-time but must never be set back after a crash. In a simple implementation, using the low order bits of a clock as the time stamp, the generation of unique identifiers is overall limited to no more than 1 per second on average. The type flags were described in Section 3.1.

An entity may migrate between hosts. Thus, an implementation can heuristically use the embedded Internet address to locate an entity but should be prepared to maintain a cache of redirects for migrated entities, plus accept Notify operations indicating that migration has occurred.

Entity group identifiers in Domain 1 are structured in one of two forms, depending on whether they are well-known or dynamically allocated identifiers. A well-known entity identifier is structured as:

```
+-----------+----------------+-----------------------+
| TypeFlags |  Discriminator |Internet Host Group Addr|
+-----------+----------------+-----------------------+
   4 bits          28 bits                32 bits
```

with the second high-order bit (GRP) set to 1. This form of entity identifier is mapped to the Internet host group address specified in the low-order 32 bits. The Discriminator distinguishes group identifiers using the same Internet host group. Well-known entity group identifiers should be allocated to correspond to the basic services provided by hosts that are members of the group, not specifically because that service is provided by VMTP. For example, the well-known entity group identifier for the domain name service should contain as its embedded Internet host group address the host group for Domain Name servers.

A dynamically allocated entity identifier is structured as:

```
+-----------+----------------+-----------------------+
| TypeFlags |  Discriminator |   Internet Host Addr   |
+-----------+----------------+-----------------------+
   4 bits          28 bits             32 bits
```

with the second high-order bit (GRP) set to 1. The Internet address in the low-order 32 bits is a Internet address assigned to the host that dynamically allocates this entity group identifier. A dynamically allocated entity group identifier is mapped to Internet host group address 232.X.X.X where X.X.X are the low-order 24 bits of the Discriminator subfield of the entity group identifier.

We use the following notation for Domain 1 entity identifiers <10> and propose it use as a standard convention.

<flags>-<discriminator>-<Internet address>

where <flags> are [X]{BE,LE,RG,UG}[A]

X = reserved
BE = big-endian entity
LE = little-endian entity
RG = restricted group
UG = unrestricted group
A = alias

and <discriminator> is a decimal integer and <Internet address> is in standard dotted decimal IP address notation.

V.1. Authentication Domain 1

A principal identifier is structured as follows.

```
+--------------------------+-----------------------+
|     Internet Address     | Local User Identifier |
+--------------------------+-----------------------+
            32 bits                   32 bits
```

VI. IP Implementation

VMTP is designed to be implemented on the DoD IP Internet Datagram Protocol (although it may also be implemented as a local network protocol directly in "raw" network packets.)

The well-known entity identifiers specified to date are:

VMTP_MANAGER_GROUP   RG-1-224.0.1.0
Managers for VMTP operations.

VMTP_DEFAULT_BECLIENT BE-1-224.0.1.0
Client entity identifier to use when a (big-endian) host has not determined or been allocated any client entity identifiers.

VMTP_DEFAULT_LECLIENT LE-1-224.0.1.0
Client entity identifier to use when a (little-endian) host has not determined or been allocated any client entity identifiers.

Note that 224.0.1.0 is the host group address assigned to VMTP and to which all VMTP hosts belong.

6.05. RFC 1146 TCP alternate checksum options

There are no IPv4 dependencies in this specification.

6.06. RFC 1151 Version 2 of the Reliable Data Protocol (RDP)

There are no IPv4 dependencies in this specification.

6.07. RFC 1644 T/TCP -- TCP Extensions for Transactions Functional Specification

There are no IPv4 dependencies in this specification.

6.08. RFC 1693 An Extension to TCP : Partial Order Service

There are no IPv4 dependencies in this specification.

6.09. RFC 1791 TCP And UDP Over IPX Networks With Fixed Path MTU

There are no IPv4 dependencies in this specification.

6.10. RFC 2343 RTP Payload Format for Bundled MPEG

There are no IPv4 dependencies in this specification.

6.11. RFC 2582 The NewReno Modification to TCP's Fast Recovery Algorithm

There are no IPv4 dependencies in this specification.

6.12. RFC 2762 Sampling of the Group Membership in RTP

There are no IPv4 dependencies in this specification.

6.13. RFC 2859 A Time Sliding Window Three Colour Marker (TSWTCM)

This specification is both IPv4 and IPv6 aware and needs no changes.

6.14. RFC 2861 TCP Congestion Window Validation

This specification is both IPv4 and IPv6 aware and needs no changes.

6.15. RFC 2909 The Multicast Address-Set Claim (MASC) Protocol

This specification is both IPv4 and IPv6 aware and needs no changes.

7.0. Summary of Results

In the initial survey of RFCs 24 positives were identified out of a total of 104, broken down as follows:

Standards:	3 out of 5 or 60.00%
Draft Standards:	0 out of 2 or 0.00%
Proposed Standards:	17 out of 82 or 20.73%
Experimental RFCs:	4 out of 15 or 26.67%

Of those identified many require no action because they document outdated and unused protocols, while others are document protocols that are actively being updated by the appropriate working groups. Additionally there are many instances of standards that SHOULD be updated but do not cause any operational impact if they are not updated. The remaining instances are documented below.

7.1. Standards

7.1.1. STD 7 Transmission Control Protocol (RFC 793)

Section 3.1 defines the technique for computing the TCP checksum that uses the 32 bit source and destination IPv4 addresses. This problem is addressed in RFC 2460 Section 8.1.

7.1.2. STD 19 Netbios over TCP/UDP (RFCs 1001 & 1002)

These two RFCs have many inherent IPv4 assumptions and a new set of protocols must be defined.

7.1.3. STD 35 ISO Transport over TCP (RFC 1006)

This problem has been fixed in RFC 2126, ISO Transport Service on top of TCP.

7.2. Draft Standards

There are no draft standards within the scope of this document.

7.3. Proposed Standards

7.3.01. TCP/IP Header Compression over Slow Serial Links (RFC 1144)

This problem has been resolved in RFC2508, Compressing IP/UDP/RTP Headers for Low-Speed Serial Links. See also RFC 2507 & RFC 2509.

7.3.02. ONC RPC v2 (RFC 1833)

The problems can be resolved with a definition of the NC_INET6 protocol family.

7.3.03. RTSP (RFC 2326)

Problem has been acknowledged by the RTSP developer group and will be addressed in the move from Proposed to Draft Standard. This problem is also addressed in RFC 2732, IPv6 Literal Addresses in URL’s.

7.3.04. SDP (RFC 2327)

One problem is addressed in RFC 2732, IPv6 Literal Addresses in URL’s. The other problem can be addressed with a minor textual clarification. This must be done if the document is to transition from Proposed to Draft. These problems are solved by documents currently in Auth48 or IESG discuss.

7.3.05. IPPM Metrics (RFC 2678)

The IPPM WG is working to resolve these issues.

7.3.06. IPPM One Way Delay Metric for IPPM (RFC 2679)

The IPPM WG is working to resolve these issues. An ID is available (draft-ietf-ippm-owdp-03.txt).

7.3.07. IPPM One Way Packet Loss Metric for IPPM (RFC 2680)

The IPPM WG is working to resolve these issues.

7.3.09. Round Trip Delay Metric for IPPM (RFC 2681)

The IPPM WG is working to resolve these issues.

7.3.08. The PINT Service Protocol: Extensions to SIP and SDP for IP Access to Telephone Call Services(RFC 2848)

This specification is dependent on SDP which has IPv4 dependencies. Once these limitations are fixed, then this protocol should support IPv6.

7.3.09. TCP Processing of the IPv4 Precedence Field (RFC 2873)

The problems are not being addressed.

7.3.10. Integrated Services in the Presence of Compressible Flows (RFC 3006)

This document defines a protocol that discusses compressible flows, but only in an IPv4 context. When IPv6 compressible flows are defined, a similar technique should also be defined.

7.3.11. SDP For ATM Bearer Connections (RFC 3108)

The problems are not being addressed, but it is unclear whether the specification is being used.

7.3.12. The Congestion Manager (RFC 3124)

An update to this document can be simply define the use of the IPv6 Traffic Class field since it is defined to be exactly the same as the IPv4 TOS field.

7.4. Experimental RFCs

7.4.1. Reliable Data Protocol (RFC 908)

This specification relies on IPv4 and a new protocol standard may be produced.

7.4.2. Internet Reliable Transaction Protocol functional and interface specification (RFC 938)

This specification relies on IPv4 and a new protocol standard may be produced.

7.4.3. NETBLT: A bulk data transfer protocol (RFC 998)

This specification relies on IPv4 and a new protocol standard may be produced.

7.4.4. VMTP: Versatile Message Transaction Protocol (RFC 1045)

This specification relies on IPv4 and a new protocol standard may be produced.

7.4.5. OSPF over ATM and Proxy-PAR (RFC 2844)

This specification relies on IPv4 and a new protocol standard may be produced.

8.0. Security Considerations

This memo examines the IPv6-readiness of specifications; this does not have security considerations in itself.

9.0. Acknowledgements

The authors would like to acknowledge the support of the Internet Society in the research and production of this document. Additionally the author, Philip J. Nesser II, would like to thanks his partner in all ways, Wendy M. Nesser.

The editor, Andreas Bergstrom, would like to thank Pekka Savola for guidance and collection of comments for the editing of this document. He would further like to thank Allison Mankin, Magnus Westerlund and Colin Perkins for valuable feedback on some points of this document.

10.0. Normative Reference

[1] Nesser, II, P. and A. Bergstrom, Editor, "Introduction to the Survey of IPv4 Addresses in Currently Deployed IETF Standards", RFC 3789, June 2004.

11.0. Authors' Addresses

Please contact the authors with any questions, comments or suggestions at:

Philip J. Nesser II
Principal
Nesser & Nesser Consulting
13501 100th Ave NE, #5202
Kirkland, WA 98034

Phone: +1 425 481 4303
Fax: +1 425 48
EMail: phil@nesser.com

Andreas Bergstrom, Editor
Ostfold University College
Rute 503 Buer
N-1766 Halden
Norway

EMail: andreas.bergstrom@hiof.no

12.0. Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.